Algorithms
Chapter 1: Algorithim Analysis

GATE CS PYQ
Solved by Monalisa

MonalisaCS

Section 5: Algorithms

- Searching, sorting, hashing. Asymptotic worst case time and space complexity. Algorithm design techniques : greedy, dynamic programming and divide-and-conquer . Graph traversals, minimum spanning trees, shortest paths
- Chapter 1:Algorithim Analysis:-Algorithm intro , Order of growth ,Asymptotic notation, Time complexity, space complexity, Analysis of Recursive & non recursive program, Master theorem]
- Chapter 2:Brute Force:-Sequential search, Selection Sort and Bubble Sort , Radix sort, Depth first Search and Breadth First Search.
- Chapter 3: Decrease and Conquer :- Insertion Sort, Topological sort,Binary Search .
- Chapter 4: Divide and conquer:-Min max problem , matrix multiplication ,Merge sort ,Quick Sort , Binary Tree Traversals and Related Properties .
- Chapter 5: Transform and conquer:- Heaps and Heap sort, Balanced Search Trees.
- Chapter 6: Greedy Method:-knapsack problem , Job Assignment problem, Optimal merge, Hoffman Coding, minimum spanning trees, Dijkstra's Algorithm.
- Chapter 7: Dynamic Programming:-The Bellman-Ford algorithm ,Warshall's and Floyd's Algorithm ,Rod cutting, Matrix-chain multiplication ,Longest common subsequence ,Optimal binary search trees
- Chapter 8: Hashing.
- Reference : Introduction to Algorithms by Thomas H. Cormen
- Introduction to the Design and Analysis of Algorithms, by Anany Levitin
- My Note

MonalisaCS

https://monalisacs.com/

- GATE CS 2010,Q12: Two alternative packages A and B are available for processing a database having 10^k records. Package A requires $0.0001n^2$ time units and package B requires $10n\log_{10}n$ time units to process n records. What is the smallest value of k for which package B will be preferred over A?
- (A) 12 (B)10 (C)6 (D)5
- B $\leq$ A
- $10\ n\ \log_{10}n \leq 0.0001n^2$
- Given n = 10^k records.
- $\Rightarrow 10\times(10^k)\log_{10}10^k \leq 0.0001(10^k)^2$
- $\Rightarrow 10^{k+1}\times k \leq 10^{-4} \times 10^{2k}$
- $\Rightarrow k \leq 10^{2k-k-1-4}$
- $\Rightarrow k \leq 10^{k-5}$
- $\Rightarrow 5 \leq 10^{5-5} = 5 \leq 1$ False
- $\Rightarrow 6 \leq 10^{6-5} = 6 \leq 10$ True
- Hence, value 5 does not satisfy but value 6 satisfies.
- 6 is the smallest value of k for which package B will be preferred over A.
- Ans : (C)6

MonalisaCS

https://www.youtube.com/@MonalisaCS

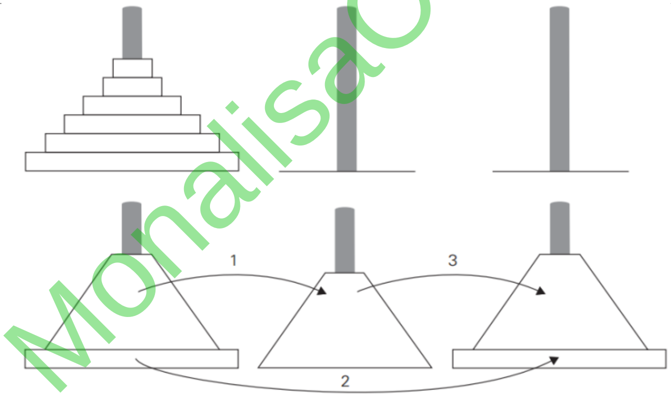
- GATE CS 2011,Q37: Which of the given options provides the increasing order of asymptotic complexity of functions f_1, f_2, f_3 and f_4?
- $f_1(n) = 2^n$ $f_2(n) = n^{3/2}$ $f_3(n) = nLog_2n$ $f_4(n) = n^{Log_2 n}$
- **(A)** f_3, f_2, f_4, f_1 **(B)** f_3, f_2, f_1, f_4 **(C)** f_2, f_3, f_1, f_4 **(D)** f_2, f_3, f_4, f_1
- Linearithmic: $f_3(n) = nLog_2n$
- Polynomial: $f_2(n) = n^{3/2}$, $f_4(n) = n^{Log_2 n}$
- Exponential: $f_1(n) = 2^n$
- $n^{3/2}$? $n^{Log_2 n}$
- 3/2 ? $\log_2 n$
- 3/2 < $\log_2 n$
- $n^{3/2}$ < $n^{Log_2 n}$
- $nLog_2n$, $n^{3/2}$, $n^{Log_2 n}$, 2^n
- Ans: **(A)** f_3, f_2, f_4, f_1

MonalisaCS

- GATE CS 2012 ,Q5:The worst case running time to search for an element in a balanced binary search tree with $n2^n$ elements is
- (A) $\Theta(n \log n)$ (B) $\Theta(n2^n)$ (C) $\Theta(n)$ (D) $\Theta(\log n)$
- Running time to search for an element in a balanced binary search tree log x
- x=number of nodes/elements
- $n2^n$= number of nodes/elements
- Running time =$\log (n2^n)$
- =$\log n + \log (2^n)$
- =$\log n+n$
- $\Theta(n)$
- Ans: (C) $\Theta(n)$

MonalisaCS

- GATE CS 2012,Q16: The recurrence relation capturing the optimal execution time of the *Towers of Hanoi* problem with *n* discs is
- (A) $T(n)=2T(n-2)+2$ (B) $T(n)=2T(n-1)+n$
- (C) $T(n)=2T(n/2)+1$ (D) $T(n)=2T(n-1)+1$
- The recurrence relation of the *Towers of Hanoi* problem with *n* discs is : $T(n)=2T(n-1)+1$
- Ans : (D) $T(n)=2T(n-1)+1$



MonalisaCS

- GATE CS 2012 ,Q18: Let W(*n*) and A(*n*) denote respectively, the worst case and average case running time of an algorithm executed on an input of size *n*. Which of the following is **ALWAYS TRUE**?
- (A) A(*n*) = Ω (W(*n*)) (B) A(*n*) = Θ (W(*n*))
- (C) A(*n*) = O (W(*n*)) (D) A(*n*) = o (W(*n*))
- A(n) ≤W(n)
- O=≤ ,o=<
- (A) A(*n*) = Ω (W(*n*)) or A(n) ≥ W(n) wrong
- (B) A(*n*) = Θ (W(*n*)) or A(n) = W(n) wrong
- (C) A(*n*) = O (W(*n*)) or A(n) ≤W(n) correct
- (D) A(*n*) = o (W(*n*)) or A(n) < W(n) wrong
- A(n) would be upper bounded by W(n) and it will not be strict upper bound as it can be same.
- Ans : (C) A(*n*) = O (W(*n*))

MonalisaCS

https://monalisacs.com/

- GATE CS 2013,Q31: Consider the following function:
- *int unknown(int n){*
- *int i, j, k=0;*
- *for (i=n/2; i≤n; i++)*
- *for (j=2; j ≤ n; j=j*2)*
- *k = k + n/2;*
- *return (k);*
- *}*
- The return value of the function is
- (A) $\Theta(n^2)$ (B) $\Theta(n^2 \log n)$ (C) $\Theta(n^3)$ (D) $\Theta(n^3 \log n)$
- The outer loop run for n/2 times and inner loop run for logn times.
- Now in each iteration k is incremented by n/2.
- So, k will be added (n/2)∗(n/2) ∗logn with an initial value of 0.
- k= $(n^2/4)$ logn : $\Theta(n^2 \log n)$
- Time complexity =n/2*logn : Θ(n logn)
- Ans : (B) $\Theta(n^2 \log n)$

i	j	k
n/2	(log n times)	(n/2) logn
n/2 +1	(log n times)	nlogn
n/2 +2	(log n times)	3(n/2) logn
…	……	…..
n	(logn times)	(n/2)*(n/2)logn

MonalisaCS

https://www.youtube.com/@MonalisaCS

- GATE CS 2014 Set 2,Q13: Which one of the following correctly determines the solution of the recurrence relation with T(1)=1?
- $T(n)=2T(\frac{n}{2})+log\ n$
- (A)Θ(n) (B)Θ(n log n) (C)$\Theta(n^2)$ (D)Θ(log n)
- Apply Master's theorem, a=2,b=2,f(n)=log n
- $n^{\log_b a} = n^{\log_2 2} = n$
- $\text{Logn} < n \Rightarrow f(n) = O(n^{\log_b a})$ case 1
- Running Time $\Theta(n^{\log_b a}) = \Theta(n)$
- Ans: (A)Θ(n)

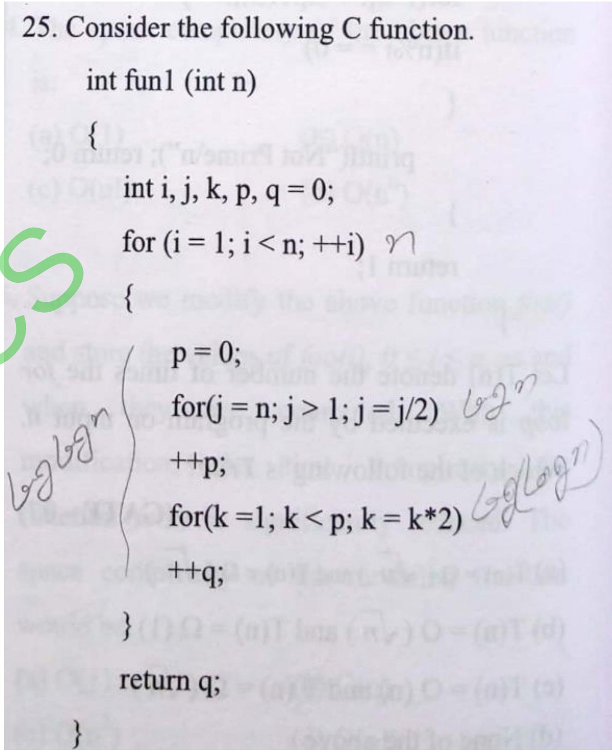
MonalisaCS

- GATE CS 2014 Set 3,Q37: Suppose you want to move from 0 to 100 on the number line. In each step, you either move right by a unit distance or you take a *shortcut*. A shortcut is simply a pre-specified pair of integers *i*,*j* with $i < j$. Given a shortcut *i*,*j* if you are at position *i* on the number line, you may directly move to *j*.Suppose *T(k)* denotes the smallest number of steps needed to move from *k* to 100. Suppose further that there is at most 1 shortcut involving any number, and in particular, from 9 there is a shortcut to 15. Let *y* and *z* be such that *T(9)*=1+min*(T(y),T(z))*. Then the value of the product yz is ______.
- T(k) = smallest number of steps needed to move from k to 100.
- T(9)=1+min(T(y),T(z)) i.e.,
- T(9)=1+min(Steps from y to 100, Steps from z to 100)
- y and z are two possible values that can be reached from 9.
- 9→10,9→15
- y=10,z=15
- yz=10*15=150
- Ans: 150

MonalisaCS

- GATE CS 2015 Set 1,Q31: Consider the following C function.

```
int fun1 (int n)
{
  int i, j, k, p, q = 0;
  for (i = 1; i<n; ++i)
  {
      p = 0;
      for (j = n; j > 1; j = j/2)
        ++p;
      for (k = 1; k < p; k = k*2)
        ++q;
  }
  return q;
}
```



25. Consider the following C function.

```
int fun1 (int n)
{
  int i, j, k, p, q = 0;
  for (i = 1; i < n; ++i)
  {
     p = 0;
     for(j = n; j > 1; j = j/2)
     ++p;
     for(k =1; k < p; k = k*2)
     ++q;
  }
  return q;
}
```

MonalisaCS

- Which one of the following most closely approximates the return value of the function fun1?
- (A) n^3 (B) $n (\log n)^2$ (C) $n \log n$ (D) $n \log(\log n)$
- *Ans D*

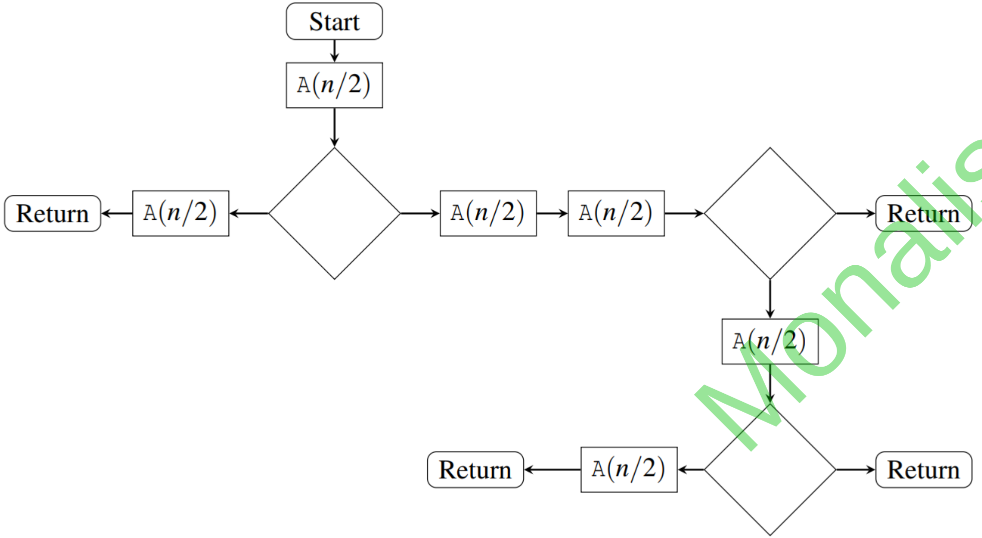
- GATE CS 2015 Set 3,Q4: Consider the equality $\sum_{i=0}^{n} i^3$=X and the following choices for X
- I.$\Theta(n^4)$ II.$\Theta(n^5)$ III.$O(n^5)$ IV.$\Omega(n^3)$
- The equality above remains correct if X is replaced by
- (A) Only I (B)Only II (C)I or III or IV but not II (D)II or III or IV but not I
- $\sum_{i=0}^{n} i^3 \cong n^4$
- I.$\Theta(n^4) \Rightarrow n^4 = n^4$,True.
- II.$\Theta(n^5) \Rightarrow n^4 \neq n^5$,False.
- III.$O(n^5) \Rightarrow n^4 \leq n^5$,True.
- IV.$\Omega(n^3) \Rightarrow n^4 \geq n^5$,True.
- (A) Only I :Wrong
- (B) Only II :Wrong
- (C)I or III or IV but not II :Correct
- (D)II or III or IV but not I :Wrong
- Ans : (C)I or III or IV but not II

MonalisaCS

- GATE CS 2015 Set 3,Q42:
- Let $f(n)=n$ and $g(n)=n^{(1+\sin n)}$, where n is a positive integer. Which of the following statements is/are correct?
- *I.* $f(n)=O(g(n))$ *II.* $f(n)=\Omega(g(n))$
- (A)Only I   (B)Only II   (C)Both I and II   (D)Neither I nor II
- The value of sine function varies from -1 to 1.
- For sin n = -1
- $g(n)=n^{(1+\sin n)} = n^{(1-1)} = n^0=1$
- $f(n)=\Omega(g(n))$
- I becomes false.
- For sin = 1
- $g(n)=n^{(1+\sin n)} = n^{(1+1)} = n^2$
- $f(n)=O(g(n))$
- II becomes false
- $f(n)=n$, $g(n)=\{1\ldots n^2\}$
- Ans : (D)Neither I nor II

MonalisaCS

- GATE CS 2016 Set 2,Q39: The given diagram shows the flowchart for a recursive function A(n). Assume that all statements, except for the recursive calls, have $O(1)$ time complexity. If the worst case time complexity of this function is $O(n^{\alpha})$, then the least possible value (accurate up to two decimal positions) of α is ______ .
- Flowchart for Recursive Function A(n)



- Total function calls 6.
- Worst case =5
- The Recurrence A(n) = 5A(n/2) + O(1)
- Apply Master's theorem
- a=5, b=2, f(n)=1,
- $n^{\log_b a} = n^{\log_2 5} = n^{2.3219280}$
- $f(n) < n^{\log_b a}$ So Case 1 ,
- Time complexity is $O(n^{2.3219280})$
- $\therefore \alpha = 2.3219280$ 0r 2.32
- Ans : 2.32

MonalisaCS

- GATE CS 2017 Set 1,Q4: Consider the following functions from positives integers to real numbers 10, √n, n, $\log_2 n$, 100/n.
- The CORRECT arrangement of the above functions in increasing order of asymptotic complexity is:
- (A) $\log_2 n$, 100/n, 10, √n, n
- (B) 100/n, 10, $\log_2 n$, √n, n
- (C) 10, 100/n ,√n, $\log_2 n$, n
- (D) 100/n, $\log_2 n$, 10 ,√n, n
- Constant , Logarithm ,Root, Linear
- 10 , $\log_2 n$, √n, n
- 100/n <10
- 100/n ,10 , $\log_2 n$, √n, n
- Ans : **(B)** 100/n, 10, $\log_2 n$, √n, n

MonalisaCS

- GATE CS 2017 Set 2,Q3:Match the algorithms with their time complexities:
- **Algorithm** | **Time Complexity**

Algorithm	**Time Complexity**
(P)Tower of Hanoi with *n* disks	(i)$\Theta(n^2)$
(Q)Binary search given *n* sorted numbers	(ii)$\Theta(n \log n)$
(R)Heap sort given *n* numbers at worst case	(iii)$\Theta(2^n)$
(S) Addition of two $n \times n$ matrices	(iv)$\Theta(\log n)$

- (A) P→(iii),Q→(iv),R→(i),S→(ii)
- (B) P→(iv),Q→(iii),R→(i),S→(ii)
- (C) P→(iii),Q→(iv),R→(ii),S→(i)
- (D) P→(iv),Q→(iii),R→(ii),S→(i)
- (P) Tower of Hanoi with n disks takes (iii)$\Theta(2^n)$ time
- (Q) Binary Search given n sorted numbers takes (iv)$\Theta(\log n)$ time
- (R) Heap sort given n numbers of the worst case takes (ii)$\Theta(n \log n)$ time
- (S) Addition of two n×n matrices takes (i)$\Theta(n^2)$ time
- Ans : (C) P→(iii),Q→(iv),R→(ii),S→(i)

MonalisaCS

- GATE CS 2017 Set 2,Q30:Consider the recurrence function
- $T(n)=2T(\sqrt{n})+1, \ n > 2$
- $T(n)=2, \quad 0 < n \leq 2$
- Then $T(n)$ in terms of Θ notation is
- (A)$\Theta(\log\log n)$ (B)$\Theta(\log n)$ (C)$\Theta(\sqrt{n})$ (D)$\Theta(n)$
- Using back substitution method
- $T(n)=2T(n^{1/2})+1$ $[T(n^{1/2})=2T(n^{1/2^2})+1]$
- $T(n)=2\times[2T(n^{1/2^2})+1]+1=2^2T(n^{1/2^2})+2+1$ $[T(n^{1/2^2})=2T(n^{1/2^3})+1]$
- $T(n)=2^2[2T(n^{1/2^3})+1]+2+1=2^3T(n^{1/2^3})+2^2+2+1$
- …………..
- $T(n)=2^iT(n^{1/2^i})+2^{i-1}+........+2^2+2+1=2^iT(n^{1/2^i})+2^i-1$ [General pattern]
- The initial condition $T(2)=2$, $n^{1/2^i}=2$
- $1/2^i \log n=\log 2=1$ [Taking Log both side]
- $2^i=\log n$
- $\log n * T(2)+\log n-1$ [After Substitution]
- $=\log n * 2+\log n-1$
- $=3\log n-1 \Rightarrow$ **$\Theta(\log n)$**
- Ans (B) Θ (Log n)

MonalisaCS

- GATE CS 2017 Set 2,Q38: Consider the following C function

```
int fun(int n) {
        int i, j;
        for(i=1; i≤n; i++) {
                for (j=1; j<n; j+=i) {
                printf("%d %d", i, j);
}}}
```

- Time complexity of fun in terms of Θ notation is
- (A) $\Theta(n\sqrt{n})$ (B) $\Theta(n^2)$ (C) $\Theta(n\log n)$ (D) $\Theta(n^2\log n)$

i	1	2	3 …	n
j	1,2,3,…(n time)	1,3,5.. (n/2 time)	1,4,7..(n/3 time)..	1(1 time)

- for each i ,j runs n/i time
- n+n/2+n/3 +….n/n =n(1+1/2+1/3+…1/n)=nlogn
- Θ(nlogn)
- Ans : (C)Θ(nlogn)

MonalisaCS

- GATE CS 2019,Q37:There are n unsorted arrays: $A_1, A_2, \ldots, A_n$. Assume that n is odd.Each of $A_1, A_2, \ldots, A_n$. contains n distinct elements. There are no common elements between any two arrays. The worst-case time complexity of computing the median of the medians of $A_1, A_2, \ldots, A_n$. is
- (A)O(n) (B)O(nlogn) (C)O(n^2) (D)Ω(n^2logn)
- The median is the middle number in a sorted, ascending or descending, list of numbers
- The time complexity to find median is O(n) in an unsorted array.
- The time complexity to find median is O(1) in an sorted array
- Total time complexity is,
- = O(n)*O(n) + O(n) = O(n^2) + O(n) ≈ O(n^2)
- Ans: (C)O(n^2)

MonalisaCS

https://monalisacs.com/

- GATE CS 2020,Q2: For parameters a and b, both of which are $\omega(1)$, $T(n)=T(n^{1/a})+1$, and $T(b)=1$. Then T(n) is
- (A) $\Theta(\log_a \log_b n)$ (B) $\Theta(\log_{ab} n)$ (C) $\Theta(\log_b \log_a n)$ (D) $\Theta(\log_2 \log_2 n)$
- $T(n)=T(n^{1/a})+1$
- Using back substitution method
- $= T(n^{1/a^2})+1+1= T(n^{1/a^2})+2$ $[T(n^{1/a})= T(n^{1/a^2})+1]$
- $= T(n^{1/a^3})+1+2= T(n^{1/a^3})+3$ $[T(n^{1/a^2})= T(n^{1/a^3})+1]$
- …..
- $T(n) = T(n^{1/a^i})+i$ [General pattern]
- Let $n^{1/a^i} = b$ $\Rightarrow 1/a^i \log n= \log b$ $[\log_b a= \frac{\log_c a}{\log_c b}]$
- $\Rightarrow a^i = \frac{\log n}{\log b} = \log_b n$ $\Rightarrow i = \log_a \log_b n$ $[a^b=n \text{ then } \log_a n=b]$
- After substitution $T(n)= T(b)+ \log_a \log_b n$
- $=1+\log_a \log_b n = \Theta(\log_a \log_b n)$
- Ans : (A) $\Theta(\log_a \log_b n)$

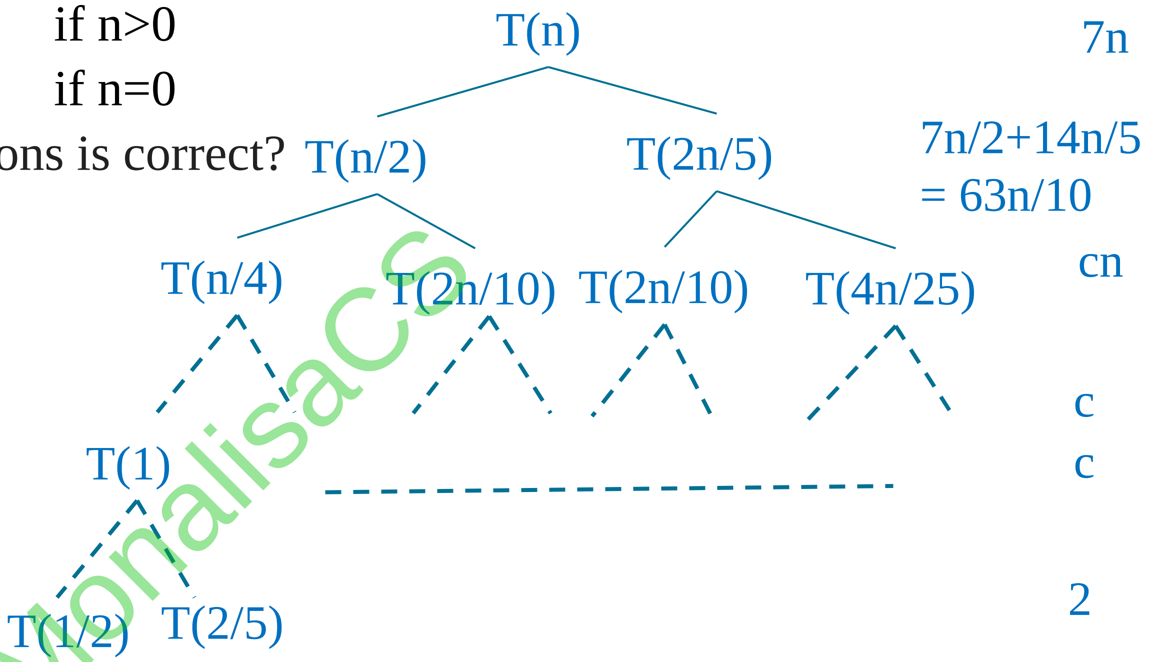
MonalisaCS

https://www.youtube.com/@MonalisaCS

- GATE CS 2021 Set 1,Q3: Consider the following three functions.
- $f_1=10^n$ $f_2=n^{logn}$ $f_3=n^{\sqrt{n}}$
- Which one of the following options arranges the functions in the increasing order of asymptotic growth rate?
- (A) f_3,f_2,f_1 (B) f_2,f_1,f_3 (C) f_1,f_2,f_3 (D) f_2,f_3,f_1
- $f_1=10^n$ Exponential Function
- $f_2=n^{logn}$ Polynomial Function
- $f_3=n^{\sqrt{n}}$ Polynomial Function
- Polynomial< Exponential
- $f_2<f_1$, $f_3<f_1$
- Compare f_2 & f_3
- $Log\ n < \sqrt{n}$
- $n^{logn} < n^{\sqrt{n}} < 10^n$
- $f_2<f_3<f_1$
- *Ans* :(D)f_2,f_3,f_1

MonalisaCS

- GATE CS 2021, Set 1,Q30:Consider the following recurrence relation.
- $T(n)=T(n/2)+T(2n/5)+7n$ if $n>0$
- $T(n)=1$ if $n=0$
- Which one of the following options is correct?
- (A) $T(n)=\Theta(n^{5/2})$
- (B) $T(n)=\Theta(n \log n)$
- (C) $T(n)=\Theta(n)$
- (D) $T(n)=\Theta((\log n)^{5/2})$



- Assume $n/2^k=1 \Rightarrow 2^k=n \Rightarrow k=\log n$
- Height =log n
- This is not a complete binary tree
- so height is not *log n* at every path.
- At lower level time will be some constant .
- cn+cn+cn+…..cn(constant times)+c+c+c… =cn
- Ans : (C) $T(n)=\Theta(n)$

MonalisaCS

- GATE CS 2021 Set 2,Q39: For constants a≥1 and b>1, consider the following recurrence defined on the non-negative integers: $T(n)=a \cdot T(\frac{n}{b})+f(n)$
- Which one of the following options is correct about the recurrence $T(n)$?
- (A) If $f(n)$ is $n\log_2(n)$, then $T(n)$ is $\Theta(n\log_2(n))$
- (B) If $f(n)$ is $\frac{n}{\log_2(n)}$, then $T(n)$ is $\Theta(\log_2(n))$
- (C) If $f(n)$ is $O(n^{\log b(a)-\epsilon})$ for some $\epsilon>0$, then $T(n)$ is $\Theta(n^{\log b(a)})$
- (D) If $f(n)$ is $\Theta(n^{\log b(a)})$, then $T(n)$ is $\Theta(n^{\log b(a)})$
- (A) wrong depends on a,b value.
- (B) wrong depends on a,b value.
- (C) If $f(n)$ is $O(n^{\log b(a)-\epsilon})$ for some $\epsilon>0$ case 1
- $T(n)$ is $\Theta(n^{\log b(a)})$,correct .
- (D) If $f(n)$ is $\Theta(n^{\log b(a)})$, then $T(n)$ is $\Theta(n^{\log b(a)} \log n)$
- But its given $\Theta(n^{\log b(a)})$, wrong.
- Ans : (C) If $f(n)$ is $O(n^{\log b(a)-\epsilon})$ for some $\epsilon>0$, then $T(n)$ is $\Theta(n^{\log b(a)})$

MonalisaCS

- GATE CS 2022 | Question: 1
- Which one of the following statements is TRUE for all positive functions $f(n)$?
- (A) $f(n^2)=\theta(f(n)^2)$, when f(n) is a polynomial
- (B) $f(n^2)=o(f(n)^2)$
- (C) $f(n^2)=O(f(n)^2)$, when $f(n)$ is an exponential function
- (D) $f(n^2)=\Omega(f(n)^2)$
- (A) $f(n^2)=\theta(f(n)^2)$ means $f(n^2)=(f(n)^2)$, when f(n) is a polynomial function.
- Let $f(n)=n^3$, $f(n^2)=(n^2)^3= n^6$ and $f(n)^2=(n^3)^2= n^6$
- So $f(n^2)=f(n)^2$ Hence True.
- (B) $f(n^2)=o(f(n^2))$ means $f(n^2) < (f(n)^2)$. This depends on f(n) hence false.
- (C) $f(n^2)=O(f(n)^2)$ means $f(n^2) \leq (f(n)^2)$, when $f(n)$ is an exponential function.
- Let $f(n)=2^n$, $f(n^2)= 2^{n^2}$ and $f(n)^2=(2^n)^2= 2^n * 2^n = 2^{2n}$
- So $f(n^2)>f(n)^2$ Hence False.
- (D) $f(n^2)=\Omega(f(n)^2)$ means $f(n^2) \geq (f(n)^2)$. This depends on f(n) hence false.
- Ans : (A) $f(n^2)=\theta(f(n)^2)$, when f(n) is a polynomial

MonalisaCS

- GATE CS 2022 | Question: 41

- Consider the following recurrence:
- $f(1)=1;$
- $f(2n)=2f(n)-1$, for $n\geq 1$;
- $f(2n+1)=2f(n)+1$, for $n\geq 1$.
- Then, which of the following statements is/are TRUE?
- (A) $f(2^n-1)=2^n-1$ (B) $f(2^n)=1$ (C) $f(5\cdot 2^n)=2^{n+1}+1$ (D) $f(2^n+1)=2^n+1$
- $f(1)=1$ $f(2)=f(2*1)=2*f(1)-1=1$
- $f(3)= f(2*1+1)=2*f(1)+1=3$ $f(4)=f(2*2)=2*f(2)-1=1$
- $f(5)= f(2*2+1)=2*f(2)+1=3$ $f(6)=f(2*3)=2*f(3)-1=5$
- $f(7)= f(2*3+1)=2*f(3)+1=7$ $f(8)=f(2*4)=2*f(4)-1=1$
- $f(9)= f(2*4+1)=2*f(4)+1=3$ $f(10)=f(2*5)=2*f(5)-1=5$
- A) for $n=2, f(2^2-1)= f(3)= 2^2-1=3$,
- for $n=3, f(2^3-1)= f(7)= 2^3-1=7$ True
- B) for $n=2, f(2^2)= f(4)= 1$, for $n=3, f(2^3)= f(8)=1$ True
- C) for $n=0, f(5.2^0)= f(5)= 2^{0+1}+1=3$,
- for $n=1$, $f(5.2^1)= f(10)= 2^{1+1}+1=5$ True
- D)for $n=0$, $f(2^0+1)= f(2)= 2^0+1=2$
- for $n=2$, $f(2^2+1)= f(5)= 2^2+1=5$ False
- Ans : A,B,C

MonalisaCS

- GATE CS 2022 | Question: 41

- Consider the following recurrence:
- $f(1)=1$; $f(2n)=2f(n)-1$, for $n\geq 1$; $f(2n+1)=2f(n)+1$, for $n\geq 1$.
- Then, which of the following statements is/are TRUE?
- (A) $f(2^n-1)=2^n-1$ (B) $f(2^n)=1$ (C) $f(5\cdot 2^n)=2^{n+1}+1$ (D) $f(2^n+1)=2^n+1$
- A) $f(2^n-1)=f(2*2^{n-1}-2+1)= f(2(2^{n-1}-1)+1)= 2f(2^{n-1}-1)+1$ [substitute $f(2^{n-1}-1)= 2f(2^{n-2}-1)+1$]
- $2*\{2f(2^{n-2}-1)+1\}+1= 2^2f(2^{n-2}-1)+2+1$
- $\ldots 2^kf(2^{n-k}-1)+2^{(k-1)}+\ldots.+2+1=2^kf(2^{n-k}-1)+2^k-1$ [Let $2^{n-k}-1 =1$, $2^{n-k}=2$, $n-k=1$, $k=n-1$]
- $2^{n-1}f(1)+2^{n-1}-1= 2^n-1$ True
- B) $f(2^n)=f(2*2^{n-1})=2 f(2^{n-1})-1$ [substitute $f(2^{n-1})= 2 f(2^{n-2})-1$]
- $2\{2 f(2^{n-2})-1\}-1=2^2 f(2^{n-2})-2-1$
- $\ldots.2^k f(2^{n-k})-2^{k-1}-\ldots.-2-1= 2^k f(2^{n-k})-\{2^k-1\}$ [Let $n-k=0$, $n=k$]
- $2^n f(2^0)-\{2^n-1\}= 2^n - 2^n +1=1$ True
- C) $f(5*2^n) = f(2*5*2^{n-1}) =2 f(5*2^{n-1})-1$ [substitute $f(5*2^{n-1})= 2 f(5*2^{n-2})-1$]
- $2\{2 f(5*2^{n-2})-1\}-1 =2^2f(5*2^{n-2})-2-1$
- $\ldots 2^kf(5*2^{n-k})- 2^{k-1}-\ldots.-2-1 = 2^kf(5*2^{n-k})- (2^k-1)$ [Let $n-k=0$, $k=n$, $f(5)=3$]
- $2^nf(5*2^0)- (2^n-1)=3*2^n - 2^n+1=2*2^n+1=2^{n+1}+1$ True
- D) $f(2^n+1)= f(2*2^{n-1} + 1)= 2f(2^{n-1})+1$ [substitute $f(2^{n-1})= 2f(2^{n-2})-1$]
- $2*\{ 2f(2^{n-2})-1\}+1=2^2*f(2^{n-2})-2+1= 2^3*f(2^{n-3})-2^2-2+1$
- $2^k f(2^{n-k})-2^{k-1}-\ldots 2^2-2+1$ [Let $n-k=0$, $k=n$]
- $2^n -2^{n-1}-\ldots 2^2-2+1= 2^n+1-(2^n-2)=3$
- Ans : A,B,C

MonalisaCS

- **GATE CS 2023 | Question: 19**
- Let f and g be functions of natural numbers given by $f(n) = n$ and $g(n) = n^2$. Which of the following statements is/are TRUE?
- (A) $f \in O(g)$ (B) $f \in \Omega(g)$ (C) $f \in o(g)$ (D) $f \in \theta(g)$
- (A) $f \in O(g) \Rightarrow n \leq C\, n^2$ True
- (B) $f \in \Omega(g) \Rightarrow n \geq C\, n^2$ False
- (C) $f \in o(g) \Rightarrow n < C\, n^2$ True
- (D) $f \in \theta(g) \Rightarrow n = C\, n^2$ False
- Ans :A,C

MonalisaCS

- **GATE CS 2023 | Question: 44**
- Consider functions **Function_1** and **Function_2** expressed in pseudocode as follows:

Function_1
```
while n > 1 do
  for i = 1 to n do
    x = x + 1;
  end for
  n = ⌊n/2⌋;
end while
```

Function_2
```
for i = 1 to 100 * n
do
    x = x + 1;
end for
```

- Let $f_1(n)$ and $f_2(n)$ denote the number of times the statement "x = x + 1" is executed in **Function 1** and **Function 2**, respectively. Which of the following statements is/are TRUE?
- (A) $f_1(n) \in \theta(f_2(n))$ (B) $f_1(n) \in o(f_2(n))$
- (C) $f_1(n) \in w(f_2(n))$ (D) $f_1(n) \in O(n)$

- **Function_1 :** Variable n becomes half for every termination of for loop
- $f_1(n) = n + \frac{n}{2} + \frac{n}{2^2} + \ldots = n(1 + \frac{1}{2} + \frac{1}{2^2} + \ldots) = \theta(n)$
- **Function_2**
- $f_2(n) = n \times 100 = \theta(n)$
- Ans : (A) $f_1(n) \in \theta(f_2(n))$, (D) $f_1(n) \in O(n)$

MonalisaCS

- **GATE DA 2024 | Question: 29**
- Consider the function **computeS** (X) whose pseudocode is given below:
- **computeS** (X)
- $S[1] \leftarrow 1$
- for $i \leftarrow 2$ to $length(X)$
-     $S[i] \leftarrow 1$
-     if $X[i - 1] \leq X[i]$
-         $S[i] \leftarrow S[i] + S[i - 1]$
-     end if
- end for
- return S
- Which **ONE** of the following values is returned by the function **computeS**(X) for $X = [6, 3, 5, 4, 10]$?
- (A) [1, 1, 2, 3, 4]   (B) [1, 1, 2, 3, 3]
- (C) [1, 1, 2, 1, 2]   (D) [1, 1, 2, 1, 5]

6	3	5	4	10
1	2	3	4	5

- S[1]=1
- for i= 2 ,S[2]=1
- If X[1] ≤ X[2] 6 ≤ 3 no,end
- for i= 3 ,S[3]=1
- If X[2] ≤ X[3] 3 ≤ 5 yes
- S[3]=1+1=2 end
- for i= 4 ,S[4]=1
- If X[3] ≤ X[4] 5 ≤ 4 no ,end
- for i= 5 ,S[5]=1
- If X[4] ≤ X[5] 4 ≤ 10 yes
- S[5]=1+1=2 end
- Ans : (C) [1, 1, 2, 1, 2]

MonalisaCS

- **GATE CS 2024 | Set 1 | Question: 7**
- Given an integer array of size N, we want to check if the array is sorted (in either ascending or descending order). An algorithm solves this problem by making a single pass through the array and comparing each element of the array only with its adjacent elements. The worst-case time complexity of this algorithm is
- **(A)** both $O(N)$ and $\Omega(N)$ **(B)** $O(N)$ but not $\Omega(N)$
- **(C)** $\Omega(N)$ but not $O(N)$ **(D)** neither $O(N)$ nor $\Omega(N)$
- We need to check whether the array is sorted or not, and it does so by making a single pass through the array.
- In worst case, the algorithm has to traverse the whole array.
- and array traversal will take O(n) time.
- **Ans : (A)** both $O(N)$ and $\Omega(N)$

MonalisaCS

- **GATE CS 2024 | Set 1 | Question: 32**
- Consider the following recurrence relation:
- $T(n) = \{\sqrt{n}T(\sqrt{n}) + n$ for $n \geq 1$, 1 for $n = 1$.
- Which one of the following options is CORRECT?
- (A) $T(n) = \Theta(n \log \log n)$ (B) $T(n) = \Theta(n \log n)$
- (C) $T(n) = \Theta(n^2 \log n)$ (D) $T(n) = \Theta(n^2 \log \log n)$
- ***Method of backward substitutions.***
- $T(n)= \sqrt{n}\ T(\sqrt{n})+n$ [substitute $T(n^{1/2})= n^{1/2^2}\ T(n^{1/2^2})+ n^{1/2}$]
- $= n^{1/2} * \{n^{1/2^2}\ T(n^{1/2^2})+ n^{1/2}\}+n= n^{3/4}\ T(n^{1/2^2})+2n$
- $= n^{1-1/2^2}\ T(n^{1/2^2})+2n$
- ………….
- General formula for the pattern: $n^{1-1/2^i}\ T(n^{1/2^i})+i*n =n/n^{1/2^i}\ T(n^{1/2^i})+i*n$
- The initial condition $T(1)=1$, $n^{1/2^i}=1$ [$\log_b a = \frac{1}{\log_a b}$]
- $1/2^i=\log_n 1 =1/\log_1 n \Rightarrow 2^i=\log n$
- $i=\log \log n$
- $n/1*T(1)+\log \log n * n=n+n \log \log n$
- **O(n log log n)**
- Ans : (A) $T(n) = \Theta(n \log \log n)$

MonalisaCS

- **GATE CS 2024 | Set 2 | Question: 5**
- Let $T(n)$ be the recurrence relation defined as follows:
- $T(0) = 1, T(1) = 2$, and
- $T(n) = 5T(n - 1) - 6T(n - 2)$ for $n \geq 2$
- Which one of the following statements is TRUE?
- (A) $T(n) = \Theta(2^n)$ (B) $T(n) = \Theta(n2^n)$ (C) $T(n) = \Theta(3^n)$ (D) $T(n) = \Theta(n3^n)$
- Method of forward Substitution
- $T(0) = 1$
- $T(1) = 2$
- $T(2) = 5T(1)-6T(0)=5*2-6*1=4=2^2$
- $T(3) = 5T(2)-6T(1)=5*4-6*2=8=2^3$
- ………..
- $T(n)=2^n$
- Ans : (A) $T(n) = \Theta(2^n)$

MonalisaCS